

图像轮廓与图像分割修复

图像修补

- src 输入图像
- inpaintMask 修复掩模
- dst 结果图像
- inpaintRadius 修补每个点的圆形领域半径
- INPAINT\_NS
- INPAINT\_TELEA
- flags 修补方法

该算法是一种基于拓扑理论的数学形态学的分割算法，经典的分水岭算法分为排序过程和标记过程两个步骤。如果将图像中的每一点像素的灰度值表示该点的海拔高度，每一个局部极小值及其影响区域称为集水盆，而集水盆的边界则形成分水岭。

分水岭算法

$E[(x-c)^k]$ 称为X关于C点的k阶矩，两种情况如下：  
1.  $c=0$ ，这时  $a_k = E[(X^k)]$ 称为X的k阶原点矩  
2.  $c=E[X]$ ，这时  $\mu_k = E[(X-E[X])^k]$ 称为X的k阶中心矩

一个从一幅数字图形中计算出来的矩集，通常描述了该图像形状的全局特征，以及该图像不同类型的集合特性信息。  
一阶矩与形状有关，二阶矩是曲线图绕直线平均值的扩展程度，三阶矩是高于平均值对称性的测量。

图像的矩

- array 输入参数
- binaryImage 若为真，非零像素均为1
- contour 输入的向量
- oriented 面向区域标识符，若为真该函数返回一个带符号的面积值
- 计算轮廓面积 `contourArea()`
- curve 输入的二维点集
- closed 是否封闭
- 计算轮廓长度 `arcLength()`
- 不变矩(Invariant Moments)是一处高度浓缩的图像特征，具有平移、灰度、尺度、旋转不变性。
- 二阶矩和三阶矩可以推出7种不变矩
- 不变矩

查找并绘制轮廓

- image 输入二值制图像，二值制图像方法
  - `compare()`
  - `inrange()`
  - `threshold()`
  - `adaptiveThreshold()`
  - `canny()`
- contours 检测到的轮廓，函数调用后的运算结果
- hierarchy 可选的输出向量，包含图像的拓扑信息
  - RETR\_EXTERNAL 只检测最外层轮廓
  - RETR\_LIST 提取所有轮廓，并且放置在list中
  - RETR\_CCOMP 提取所有轮廓，并且将其组织为双层结构
  - RETR\_TREE 提取所有轮廓，并重新建立树状的轮廓结构
- mode 轮廓检索模式
  - CHAIN\_APPROX\_NONE
  - CHAIN\_APPROX\_SIMPLE
  - CHAIN\_APPROX\_TC89\_L1
  - CHAIN\_APPROX\_TC89\_KCOS
- method 轮廓的近似方法
- offset 每个轮廓的可选偏移量
- image 目标图像
- contours 输入的所有轮廓
- contourIdx 轮廓的绘制指示变量
- color 绘制轮廓的颜色
- thickness 轮廓的粗细
- lineType 线条的类型
- hierarchy 可选的层次信息
- maxLevel 表示用于绘制轮廓的最大等级
- offset 可选的轮廓偏移参数

寻找物体的凸包

- 凸包
  - 给定二维平面上的点集，凸包就是将最外层的点连接起来构成的凸多边形。
- 寻找凸包 `convexHull()`
  - points 输入的二维点集
  - hull 输出参数，函数调用后找到的凸包
  - clockwise 操作方向标识符
  - returnPoints 操作标识符

使用多边形将轮廓包围

- 返回外部矩形边界 `boundingRect()`
- 寻找最小包围矩形 `minAreaRect()`
- 寻找最小包围圆形 `minEnclosingCircle()`
- 用椭圆拟合二维点集 `fitEllipse()`
- 逼近多边形曲线 `approxPolyDP()`
  - curve 输入的二维点集
  - approxCurve 多边形逼近的结果
  - epsilon 逼近的精度
  - closed 若为真，则近似的曲线为封闭的曲线